Paper ID: 26419
oral

SPACE SYSTEMS SYMPOSIUM (D1)
Space Systems Architectures (4)

Author: Mr. Edward Birrane
The Johns Hopkins University Applied Physics Laboratory, United States

Dr. Daisy Zhang
The John Hopkins University Applied Physics Laboratory, United States
Mr. Robert Berardino
The Johns Hopkins University Applied Physics Laboratory, United States

A SYSTEM-INDEPENDENT GROUND SOFTWARE APPROACH FOR TELEMETRY DISTRIBUTION

Abstract

Ground segments visualize, analyze, fuse, condition, and otherwise process telemetry received from spacecraft. Dedicated server software, referred to as Command and Telemetry (CT) Cores, feed and manage telemetry archives. The diversity of commercially available CT Core products provides flexibility for missions of varying size and budget, but complicates the engineering of re-usable software applications. Individual CT Cores provide different features, have different Application Programming Interfaces, and are written in different programming languages; they may be deployed across hardware platforms with incompatible bit representations, data precision, and operating systems. Missions often must balance the cost savings of a less-expensive CT Core with the added expense of refactoring ground software applications, even when these applications represent mission-independent capabilities. Avoiding refactors dramatically reduces software development, testing, training, and maintenance while allowing missions to contain costs by selecting the CT Core that best matches their budget. Applications operating on mission telemetry must be built on an abstracting layer that hides the underlying CT Core. Implementing that layer as a messaging protocol insulates software from incompatibilities in hardware and operating systems.

This paper presents work performed by The Johns Hopkins University Applied Physics Laboratory to develop a Telemetry Request Protocol (TRP) and build protocol-driven ground software applications for telemetry distribution. We demonstrate the efficiency and flexibility of the TRP, as part of a ground architecture, in requesting telemetry of various types and formats common to our NASA missions. We describe the desirable properties, architecture, and messages of the TRP and show examples of its deployment as a standard way of communicating amongst telemetry providers and consumers. We quantify performance benefits from using binary representations (Google Protocol Buffers) versus ASCII data (XML). We discuss the implementation of this protocol and its associated servers, clients, and applications in support of NASA missions such as the Van Allen Probes and Solar Probe Plus. Additionally, we discuss lessons learned from this standardization and implementation effort to include time representations and sorting, protocol streaming performance, state transitions, and error handling.

We conclude that targeting standardization to ground system functions, such as telemetry retrieval, provides opportunity for incremental application and thus wider adoption. We observe that our implementation of the TRP proves the price/performance of this approach and demonstrates the feasibility of cross-mission, cross-product reuse. Finally, we discuss the implications of protocols as a model for incrementally standardizing other ground segment functions.